1

CONFLICT DETECTION AND RESOLUTION IN ASSOCIATION WITH DATA ALLOCATION

TECHNICAL FIELD OF THE INVENTION

This invention relates to the field of data allocation, and more particularly to conflict detection and resolution in association with data allocation.

BACKGROUND OF THE INVENTION

It is often desirable within a business or other planning environment to generate information regarding demand, available supply, selling price, or other data concerning a product or other item. Data for products may often be dependent in some manner on data for other hierarchically related products. For example, demand for a product with multiple components may drive the demand for a particular one of those components. Similarly, demand for products in a particular geographic region may drive the demand for the products in a particular territory in the region. Because of these hierarchical dependencies, the data concerning various products or other items may be stored hierarchically in data storage or derived in a hierarchical fashion. Furthermore, the data may be stored at a storage location associated with multiple dimensions, such as a product dimension (the storage location being associated with a particular product or product component), a geography dimension (the storage location being associated with a particular geographical area), and a time dimension (the storage location being associated with a particular time or time period).

It is often desirable to update product data by forecasting demand values or other appropriate values for a particular product or group of products. As an example, using the current and/or past demand values associated with a particular product in a particular state, the demand for the product in that state at a time in the future may be forecasted. However, it may not be feasible or accurate to forecast demand values for the product in a particular region of the state or to forecast demand values for the product in the entire country in which the state is included. Instead, the demand value for the product in the particular state may be used to determine other hierarchically related demand values using allocation techniques. For example, the forecasted demand value may be determined by aggregating it with demand values for the product in other states in the country to determine a demand value for the product in the entire country. Alternatively, the demand value may be allocated by disaggregating it to determine a demand value for the product in each of the regions of the state. However, many current allocation methods do not provide a sufficiently accurate allocation of forecasted values and thus negatively affect demand planning, supply planning, or other planning based on the allocated values.

SUMMARY OF THE INVENTION

According to the present invention, disadvantages and problems associated with previous data allocation techniques have been substantially reduced or eliminated.

5 According to one embodiment of the present invention, a method for detecting and resolving conflicts in association with a data allocation includes determining a relationship between each of a plurality of positions in a hierarchical organization of data. The method also includes selecting a position i and determining a total weight of position i. If the total weight of position i is effectively non-zero, the method 10 further includes removing the influence of position i from the other positions and adding position i to a set of conflict-free positions. Alternatively, if the total weight of position i is effectively zero, the method includes selecting a position k with which position i has a relationship, reintroducing the effect of position k on the other positions if k is already in the conflict-free set, removing position k from the conflict-15 free set if k is already in the conflict-free set, removing the influence of position i from the other positions if i is not the selected position, and adding position i to the conflict-free set if i is not the selected position. The method also includes successively repeating the method for each position, with each successive position becoming position i.

20 Embodiments of the present invention may provide one or more technical advantages. For example, particular embodiments may be used in conjunction with an allocation of values to detect and resolve potential conflicts associated with the allocation. When a complex allocation is to be performed, it may be difficult to identify the various hierarchical dependencies that may lead to conflicts. Certain 25 embodiments may ensure that when a set of positions in an hierarchical organization of data are selected from which to allocate values to lower level positions in the hierarchy, the selected positions will be free from conflicts that might make the allocation infeasible. Furthermore, particular embodiments may use techniques for identifying such conflicts that generate values that may be used when actually 30 performing the allocation. Therefore, many of the computations performed during a conflict identification and resolution process by particular embodiments may be re-

used in the associated allocation and thus the conflict identification and resolution process is computationally efficient.

Other important technical advantages are readily apparent to those skilled in the art from the figures, descriptions, and claims included herein.

BRIEF DESCRIPTION OF THE DRAWINGS

To provide a more complete understanding of the present invention and the features and advantages thereof, reference is made to the following description taken in conjunction with the accompanying drawings, in which:

5   FIGURE 1 illustrates an example system for allocating data in a hierarchical organization of data;

FIGURE 2 illustrates an example product dimension within a multi-dimensional organization of data;

FIGURE 3 illustrates an example geography dimension within a multi-dimensional organization of data;

10

FIGURE 4 illustrates an example method for allocating data within a business or other planning environment;

FIGURE 5 illustrates an example allocation of a forecasted value associated with a single parent in one dimension;

15   FIGURE 6 illustrates an example allocation of forecasted values associated with multiple parents associated with multiple dimensions; and

FIGURE 7 illustrates an example method of detecting and resolving conflicts before an allocation.

DESCRIPTION OF EXAMPLE EMBODIMENTS

FIGURE 1 illustrates an example system 10 for allocating data, such as forecasted data, in a hierarchical organization of data associated with a business or other planning environment. As described below, system 10 implements an allocation

5   strategy that may be used to allocate a value associated with a particular data member in a data storage device or in a representation of data to hierarchically related data members. In general, forecasted data provides an estimate of the forecasted demand, available supply, selling price, or other quantifiable data measure associated with a particular product or other item. Although products are typically referred to herein,

10   system 10 may be used to allocate forecast data for appropriate tangible or non-tangible items other than products, including but not limited to services or other benefits. Furthermore, although forecasts are primarily discussed herein, system 10 may also allocate historical or other data, separately or in combination with forecast data, according to particular planning needs of an enterprise, facility, or user.

15   Moreover, although the allocation of demand forecasts for products is primarily described, those skilled in the art will appreciate that system 10 may also allocate forecasts for available supply, selling price, and any other suitable data.

System 10 includes client 12, server 14, and data storage 16. Client 12 may include one or more processes to provide appropriate administration, analysis, and

20   planning input. Although these processes are preferably separate processes running on a dedicated client processor, these processes may be integrated, in whole or in part, and run on one or more processors within the same or different computers. Similarly, the server 14 may include one or more processes to receive administration, analysis, and planning input from client 12 and interact with data storage 16 to provide

25   corresponding output to client 12. Although the processes are preferably separate processes running on a dedicated server processor, these processes may be integrated, in whole or in part, and run on one or more processors within the same or different computers. Client 12 and server 14 may be fully autonomous or may operate at least in part subject to input from users of system 10.

30   The term "data storage" is used to refer to any appropriate data source, representation of data, or other organization of data. Data storage 16 may be hierarchical in nature, may be multi-dimensional, and/or may provide persistent data

storage for system 10. For example, data storage 16 may be a multi-dimensional database that stores data in a hierarchical and multidimensional format or data storage 16 may be a representation of data derived by server 12 or other appropriate component from data stored in a relational database, in memory, or in any other

5   appropriate location. Server 12 or other appropriate component may use a multi-dimensional hierarchical transformation layer to create such a representation of the data. In one embodiment, data storage 16 includes three-dimensional data and, for each data measure, associates with each storage location 18 a particular member from the product dimension, a particular member from the geography dimension, and a

10 particular member from the time dimension. Each of these particular combinations of members of these three dimensions is associated with a corresponding storage location 18 in data storage 16, similar to each combination of coordinates from the x, y, and z axes being associated with a point in three-dimensional Euclidian space. Furthermore, position within a particular dimension may be changed independent of

15 members of other dimensions, much like the position of a coordinate along the x axis may be changed independent of the positions of other coordinates along the y and z axes in three-dimensional Euclidian space.

Data storage 16 may have as few or as many dimensions as appropriate for the particular application. For example, and not by way of limitation, an enterprise

20 associated with system 10 may not consider geography in connection with its data forecasting needs. This might be the case when products are ordered using the Internet or the telephone and then distributed from a single distribution point. In this example, data storage 16 might be two-dimensional rather than three-dimensional and might not reflect positions or members within the geography dimension.

25 Furthermore, the demand or other data might be quantified per specified time interval, in which case data storage 16 might be two-dimensional and might not reflect positions or members within the time dimension. Other possible scenarios involving more or fewer than three dimensions will be apparent to those skilled in the art. Data storage 16 may have any number of dimensions appropriate for the needs of the

30 enterprise or facility associated with system 10.

In the three-dimensional case, the values of the data measures within the set for a particular storage location 18 depend on the combined positions of members

within product, geography, and time dimensions for that storage location 18. As a result, the values of the data measures typically vary with these combined positions as appropriate to accurately reflect the demand, available supply, selling price, or other data associated with these members. As described below, when a suitable

5   combination of members is specified in the product, geography, and time dimensions according to operation of system 10, data storage 16 accesses the data measures for storage location 18 associated with that combination of members to assist system 10 in allocating demand forecasts or other suitable data. Other suitable dimensions may replace or be combined with the product, geography, and time dimensions according

10 to particular needs.

In one embodiment, data storage 16 supports multi-dimensional on-line analytical processing (OLAP) capability and is populated with data measures received from one or more transactional data sources that are internal, external, or both internal and external to the enterprise or facility associated with system 10. For example, and

15 not by way of limitation, data measures received from sources internal to a manufacturing or warehousing facility may include unit shipping data, dollar shipping data, inventory data, pricing data, and any other suitable information applicable to demand forecasting. Data measures received from external sources, such as from syndicated partners of the enterprise or facility, may include point-of-sale

20 demographic data and any other suitable information. Appropriate data measures may be stored in data storage 16 in any suitable manner.

Server 12 is coupled to data storage 16 using link 32, which may be any wireline, wireless, or other link suitable to support data communications between server 12 and data storage 16 during operation of system 10. Data storage 16 may be

25 integral to or separate from server 12, may operate on one or more computers, and may store any information suitable to support the operation of system 10 in allocating demand forecasts or other data. Server 12 is coupled to client 14 using link 30, which may be any wireline, wireless, or other link suitable to support communications between server 12, client 14, and the processes of server 12 and client 14 during

30 operation of system 10. Although link 30 is shown as generally coupling server 12 to client 14, processes of server 12 may communicate directly with one or more corresponding processes of client 14.

System 10 may operate on one or more computers 20 that are integral to or separate from the hardware and software that support server 12, client 14, and data storage 16. Computer 20 may include a suitable input device 22, such as a keypad, mouse, touch screen, microphone, or other device to input information. An output device 24 may convey information associated with the operation of system 10, including digital or analog data, visual information, or audio information. Computer 20 may include fixed or removable storage media, such as magnetic computer disks, CD-ROM, or other suitable media to receive output from and provide input to system 10. Computer 30 may include one or more processors 26 and associated memory to execute instructions and manipulate information according to the operation of system 10. Although only a single computer 20 is shown, server 12, client 14, and data storage 16 may each operate on separate computers 20 or may operate on one or more shared computers 20. Each of the one or more computers 20 may be a work station, personal computer (PC), network computer, personal digital assistant (PDA), wireless data port, or any other suitable computing device.

FIGURE 2 illustrates an example product dimension 50 within data storage 16 that includes a hierarchy of product levels 52 each having one or more members 54. The value of each data measure associated with a member 54 is an aggregation of the values of corresponding data measures associated with hierarchically related members 54 in lower levels 52 of product dimension 50. In an example embodiment in which system 10 provides demand forecasts, the demand associated with a member 54 is the aggregate demand for these hierarchically related members 54 in lower levels 52 of product dimension 50. In the illustrated embodiment, product levels 52 for product dimension 50 include an all products level 58, a product type level 60, a product category level 62, and a product family level 64. Selected and merely example hierarchical relationships between members 54 are shown using links 56, as described more fully below. Links 56 between hierarchically related members 54 in adjacent levels 52 of product dimension 50 reflect parent-child relationships between members 54. Although FIGURE 2 is described primarily in connection with demand relationships, the following description is similarly applicable to other data relationships, such as available supply, selling price, or any other relationships relating to data measures associated with an item or set of items.

In the particular example shown in FIGURE 2, all products level 58 contains "All" member 54 representing the aggregate demand for all members 54 in lower levels 60, 62, and 64 of product dimension 50. Product type level 60 contains "Components," "Base Units," and "Options" members 54. "Components" member 54

5   represents the aggregate demand for hierarchically related members 54 below "Components" member 54 in levels 62 and 64 of product dimension 50. Similarly, "Base Units" member 54 represents the aggregate demand for hierarchically related members 54 below "Base Units" member 54 and "Options" member 54 represents the aggregate demand for hierarchically related members 54 below "Options" member 54.

10   Links 56 between "All" member 54 and "Components," "Base Units," and "Options" members 54 indicate the hierarchical relationships between these members 54.

Product category level 62 contains, under "Components" member 54, "Hard Drives," "Memory Boards," and "CPUs" members 54. "Hard Drives" member 54 represents the aggregate demand for hierarchically related members 54 below "Hard

15   Drives" member 54 in level 64 of product dimension 50. Similarly, "Memory Boards" member 54 represents aggregate demand for hierarchically related members 54 below "Memory Boards" member 54 and "CPUs" member 54 represents the aggregate demand for hierarchically related members 54 below "CPUs" member 54. Links 56 between "Components" member 54 and "Hard Drives," "Memory Boards,"

20   and "CPUs" members 54 indicate the hierarchical relationships between these members 54. Analogous links 56 reflect hierarchical relationships between "Base Units" and "Options" members 54 of product type level 60 and corresponding members 54 in lower levels 62 and 64 within product dimension 50.

Product family level 64 contains, under "Hard Drives" member 54, "4GB" and

25   "6GB" members 54. Links 56 between "Hard Drives" member 54 and "4GB" and "6GB" members 54 indicate hierarchical relationships between these members 54. Analogous links 56 reflect hierarchical relationships between "Memory Boards," "CPUs," "Servers," "Desktops," "Laptops," "Monitors," "Keyboards," and "Printers" members 54 of product category level 62 and corresponding members 54 in lower

30   level 64 within product dimension 50. Although no links 56 are shown between members 54 in product family level 64 and possible lower levels 52, such further levels 52 may exist within product dimension 50 and analogous links 56 may exist to

reflect the corresponding hierarchical relationships. Furthermore, members 54 shown in FIGURE 2 are example only and are not intended to be an exhaustive set of all possible members 54. Those skilled in the art will appreciate that other suitable members 54 and associated links 56 may exist.

FIGURE 3 illustrates an example geography dimension 70 within data storage 16 that includes a hierarchy of geography levels 72 each having one or more members 74. The value of each data measure associated with a member 74 is an aggregation of the values of corresponding data measures associated with hierarchically related members 74 in lower levels 72 of geography dimension 70. In the example embodiment in which system 10 provides demand forecasts, the demand associated with a member 74 is the aggregate demand for these hierarchically related members 74. In this embodiment, geography levels 72 for geography dimension 70 include a world level 78, a country level 80, a region level 82, and a district level 84. Selected and merely example hierarchical relationships between members 74 are shown using links 76, which are analogous to links 56 described above with reference to FIGURE 2. Although FIGURE 3 is described primarily in connection with demand relationships, the following description is similarly applicable to other data relationships, such as available supply, selling price, or any other relationships relating to one or more data measures associated with an item or set of items.

In the particular example illustrated in FIGURE 3, world level 78 contains "World" member 74 representing aggregate worldwide demand. Country level 80 contains "U.S." and "Canada" members 74, which represent aggregate demand for the United States and Canada, respectively. Link 76 between "U.S." members 74 in country level 80 and "World" members 74 in world level 78 indicates a hierarchical relationship between these members 74. Similarly, link 76 between "Canada" member 74 and "World" member 74 indicates a hierarchical relationship between these members 74. In this example, worldwide demand is an aggregation of aggregate demand in the United States as well as aggregate demand in Canada. Although other links 76 are not described in detail, those skilled in the art will appreciate that links 76 are analogous to links 56 described above with reference to FIGURE 2 in that each represents a corresponding hierarchical relationship between members 74 in the various levels 72 of geography dimension 70. As discussed above,

geography dimension 70 may be eliminated or otherwise not considered in allocating data, for example, if geography dimension 70 is not relevant to particular data forecasting needs. Data storage 16 might in this situation be two-dimensional.

Demand or other forecasts may be derived using traditional forecasting

5   techniques and suitable information concerning products, geographic areas, customers, and/or other data dimension. Such information may include historical sales, causal factors, key account input, market·intelligence, and the like. Forecasting techniques may rely on hierarchical relationships between members 54, 74 to allocate data forecasts for products corresponding to members 54, 74. As described above, the

10   data measures associated with each member 54, 74 are an aggregation of the data measures associated with some or all members 54, 74 in lower levels 52, 72 within the same hierarchy of parent-child links 56, 76. Therefore, given forecast data for a member 54, 74 (a parent) at one level 52, 72, the forecasts for each of the related members 54 in the next lowest level 52, 72 (the children of the parent) may be

15   determined by disaggregating the forecast data for the parent between the children. Furthermore, although the terms "parent" and "children" are used above to identify a relationship between members 54, 74 of a single dimension 50, 70, these terms may also be used to refer to the relationship between data measures or values associated with a storage location 18 associated with a member from each of a number of

20   dimensions. For example, a storage location 18 that includes a demand value for a particular product in a particular state may be hierarchically related to a storage location 18 that includes a demand value for the product in a city of that state (the value associated with the former storage location 18 being a parent of the value associated with the latter storage location 18).

25   When allocating a forecast from one or more parents to their children, a "top-down" proportional allocation strategy is often used. In this strategy, the value of the forecast (such as a demand forecast) associated with a parent is divided proportionally among its children according to the relative current values (such as current demand values) associated with the children. Therefore, using such proportional allocation,

30   children having larger values get a larger share of the number being allocated and children having smaller values get a proportionately smaller share. For example, if a parent with a forecasted demand of 1800units has a first child that currently has an

associated demand of 1000units and a second child that currently has an associated demand of 500units, then 1200units of the forecasted demand would be allocated to the first child and 600units of the forecasted demand would be allocated to the second child.

5   Top-down allocation, proportional or otherwise, may be used for many reasons. For example, forecasts that are estimated at a higher level 52, 72 often are more accurate and a forecast planner may want to keep those values intact and adjust the forecasts at the lower levels 52, 72 to agree with the higher level forecast. Alternatively, the forecasts at a high level 52, 72 may be specified, such as objectives

10   and targets, and the lower level forecasts are adjusted to achieve the target forecast at the higher level 52, 72. However, proportional allocation is often too restrictive and may adversely affect the accuracy of the forecast values determined for children in the lower level 52, 72. For example, a scenario where proportional allocation may create inaccurate forecasts is when the value associated with a child to which an estimated

15   forecast is to be allocated has a relatively high variance (for example, the value varies widely over time). In this case, a proportional allocation based on the current value associated with the child (or based on an average of a selected number of past values) may be skewed by a temporary fluctuation in the value.

        System 10 may use an allocation strategy that accounts for variance in the

20   values associated with children when allocating a forecasted value from a parent of the children. Furthermore, it is possible for the values of the children to have positive or negative relationships between themselves, so that a higher value associated with one child may have a correspondence with a higher or lower value associated with another child. The allocation strategy may also account for these relations.

25   A distance measure may be defined as follows in order to take into account the variance and correspondence between children when allocating a forecast:

$$d = (\bar{x} - \bar{x}') \; \Sigma^{-1} \; (\bar{x} - \bar{x}') \qquad\qquad (1)$$

30   In this equation, $\bar{x}$ is the vector of current values (such as demand values) associated with the children of a particular parent. Σ is the variation matrix that identifies the

variation of each child (Σ^{-1} being the inverse of the variance matrix). The variation of a particular child may be expressed as a standard deviation, variance, or any other suitable measure and may be determined using statistical equations, models, or any other appropriate technique. $\bar{x}'$ is the vector of the values associated with the children after the allocation of the forecast from the parent. To optimally allocate the forecast, the selection of the values of $\bar{x}'$ should minimize the distance d.

The determination of $\bar{x}'$ may be subject to the constraint of the parent-child relationships. For general linear relationships, such as when the value associated with a parent equals the sum or average of the values associated with its children, it is possible to define a suitable parent-child relationship matrix R such that if $\bar{y}$ is the vector of values associated with one or more parents of the children represented in $\bar{x}$ and $\bar{x}'$, then the parent-child relationship can be expressed as follows:

$$R\,\bar{x}' = \bar{y} \tag{2}$$

It should be noted that a child may have multiple parents in the same dimension or in multiple dimensions. This concept is described below with reference to FIGURE 6. Given the above two equations, an optimal $\bar{x}'$ may be given by the following equation:

$$\bar{x}' = \bar{x} + \Sigma R^T (R \Sigma R^T)^{-1} (\bar{y} - R\,\bar{x}) \tag{3}$$

where R^T is the transpose of R.

As an example only and not by way of limitation, consider a local hierarchy with one parent and three children for a time series of length T. The values of each child i may be denoted by a separate column vector $(x_{i,1}, \ldots, x_{i,T})$ and the values of the parent may be denoted by a single column vector $(y_1, y_2, \ldots, y_T)$. A single column vector including the values of all children i for all times t may be expressed as follows:

$$\bar{x} = \begin{bmatrix} x_{1,1} \\ x_{2,1} \\ x_{3,1} \\ x_{1,2} \\ x_{2,2} \\ x_{3,2} \\ \vdots \\ x_{1,T} \\ x_{2,T} \\ x_{3,T} \end{bmatrix}$$

Assuming that an example parent-child relationship indicates that, at each time, the sum of values of all the children should equal the value of the parent, the 5 parent-child relationship matrix may be expressed as follows:

$$R = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ & & & & & & \ddots & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 1 & 1 \end{bmatrix}$$

In general, the matrix **Variation(x)** representing the variations of all children 10 at all times is a square matrix of dimension equal to the product of the number of children and T. In such a matrix, most elements are equal to zero. For example, assume that the values of $\bar{x}$ are predictions from a model. **Variation(x)** assumes a typical block-diagonal structure as follows:

15 $$Variation(x) = \begin{bmatrix} \Sigma^1 & 0 & \cdots & 0 \\ 0 & \Sigma^2 & \cdots & 0 \\ & & \ddots & \\ 0 & 0 & \cdots & \Sigma^T \end{bmatrix}$$

where, the zeros represent variation matrices of all elements zero of appropriate order and each Σ^t (not italicized in the above matrix) is of the general form (for the example case of three children):

$$\Sigma^t = \begin{bmatrix} \sigma_{1,1}^t & \sigma_{1,2}^t & \sigma_{1,3}^t \\ \sigma_{2,1}^t & \sigma_{2,2}^t & \sigma_{2,3}^t \\ \sigma_{3,1}^t & \sigma_{3,2}^t & \sigma_{3,3}^t \end{bmatrix}$$

where $\sigma_{i,i}^t$ is the variation (such as the variation) of a particular child i at time t and $\sigma_{i,j}^t$ is the correlated variation or "covariation" (such as the covariance) between two different children i and j at time t. The variations and covariations may be determined using any appropriate methods, including but not limited to standard statistical formulas or more complex statistical models. In certain embodiments, the covariations are not utilized and are replaced by zeros in the above matrix.

After algebraic manipulation of the expression for $\bar{x}^t$ described above, the allocation to each child i for a particular time t amounts to adding to its value $\bar{x}$ at time t a proportion, $\dfrac{\sum_j \sigma_{i,j}^t}{\sum_{i,j} \sigma_{i,j}^t}$, of the difference at the time t between the value associated with the parent and the sum of the values associated with children. Under the most common scenario of univariate modeling and forecasting methods, the adjustment proportion would take the simpler form, $\dfrac{\sigma_{i,i}^t}{\sum_i \sigma_{i,i}^t}$.

Unlike previous allocation techniques, system 10 accounts for the variation of the data values associated with a child in a hierarchical organization of data when allocating a value to that child from a parent. Therefore, system 10 provides a more accurate allocation. Furthermore, system 10 may also take into account parent-child relationships involving different dimensions of data storage 16 when allocating a value. In addition, system 10 typically does not require complex computations to perform an allocation. Example allocation scenarios are described below with reference to FIGURES 5 and 6.

FIGURE 4 illustrates an example method for allocating data, such as forecast data, in hierarchical organization of data associated with a business or other planning environment. The method begins at step 102 where the value of one or more parents at a time t is forecasted or otherwise determined. As described above, any appropriate

5   technique for generating a forecast for a particular value (such as a demand value) associated with a parent may be used. Values associated with multiple parents in different dimensions within data storage 16 may be determined and those values may be allocated to the children of those parents according to the present method. If there is a single parent value, then this value is represented in Equations (1), (2), and (3)

10   above as a value y (instead of a vector $\bar{y}$). If there are multiple parent values, those parent values are represented in the equations as a column vector $\bar{y}$. At step 104, the current values of the children (or the values otherwise existing before allocation of the forecasted parent value) are determined. These values are represented in Equations (1), (2), and (3) as the column vector $\bar{x}$.

15   At step 106, the parent-child relationship matrix R is determined. As described above, the parent-child relationship matrix is formulated such that the value of a parent or parents at a particular time is equal to the product of the parent-child relationship matrix and the vector of the child values at that time. The variation matrix Σ for the children at the relevant time t is determined at step 108. As

20   described above, the variations included in the variation matrix may be determined using any appropriate methods. At step 110, the values of the children at time t are determined according to Equation (3) presented above. In this equation, the values of the children at time t are represented as the vector $\bar{x}'$ and are determined using the values of $\bar{y}, \bar{x}$, R, and Σ that were determined in steps 102, 104, 106, and 108,

25   respectively. However, it should be understood that these values may be determined in any appropriate order and in any appropriate manner.

FIGURE 5 illustrates an example allocation of a forecasted value associated with a single parent 200 in one dimension using the method of FIGURE 4. In this example, the current value (such as a demand value) associated with parent 200,

30   which may represent a product category C1 in product dimension 50, is 600units. Parent 200 has a first child 210a representing a product family F1 and having a

current associated value of 100units, a second child 210b representing a product family F2 and having a current associated value of 200units, and a third child 210c representing a product family F3 and having a current associated value of 300units. These values may be expressed in a vector as follows:

5

$$\bar{x} = \begin{bmatrix} 100 \\ 200 \\ 300 \end{bmatrix}$$

In this example, the sum of the values of children 210 equals the value of parent 200. Therefore, the parent-child relationship matrix may be expressed as follows:

10

$$R = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

Furthermore, assume that the variation matrix for children 210 is as follows:

15

$$\Sigma = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 25 & 0 \\ 0 & 0 & 10 \end{bmatrix}$$

Assuming that the forecasted value y associated with parent 200 at time t is 700units, the values allocated to children 210 for time t using Equation (3) above may be determined as follows:

20

$$\bar{x}' = \bar{x} + \Sigma R^T (R \Sigma R^T)^{-1} (y - R \bar{x})$$

$$\bar{x}' = \begin{bmatrix} 100 \\ 200 \\ 300 \end{bmatrix} + \begin{bmatrix} 5 \\ 25 \\ 10 \end{bmatrix} (0.025) \left(700 - \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 100 \\ 200 \\ 300 \end{bmatrix} \right)$$

$$\bar{x}' = \begin{bmatrix} 112.5 \\ 262.5 \\ 325 \end{bmatrix}$$

It should be noted that the sum of these allocated values equals the forecasted parent value. Furthermore, these values may be compared to the values obtained using a proportional allocation technique. Using such a technique, the values of the first, second, and third children 210 would be 116.67units, 233.33units, and 350units, respectfully. However, these values do not account for the variations in the values associated with each child 210 and thus likely to be less accurate than the values that are obtained above using the example method.

FIGURE 6 illustrates an example allocation of forecasted values associated with multiple parents 220 in multiple dimensions using the method of FIGURE 4. In this example, a first parent 220a is associated with a territory T1 in geography dimension 70 and also with a product category C1 in product dimension 50. The product category C1 includes two families F1 and F2. Furthermore, a second parent 220b is associated with a district D1 in geography dimension 70 and also with family F2 in product dimension 50. District D1 includes territories T1, T2, and T3. As can be seen, the first and second parents 220 each represent values (such as demand values) associated with two dimensions. Parent 220a represents values associated with product category C1 in territory T1. Parent 220b represents values associated with district D1 for product family F2.

Parent 220a has a first child 230a that represents values associated with product family F1 in territory T1 and has a second child 230b that represents values associated with product family F2 in territory T1. Parent 220b has a first child 230b that represents values associated with product family F2 in territory T1, a second child 230c that represents values associated with product family F2 in territory T2, and a third child 230d that represents values associated with product family F2 in territory T3. Therefore, parents 220 share a single child 230b representing values associated with product family F2 in territory T1.

In this example, the current value (such as a demand value) associated with parent 220a is 300units. Child 230a has an associated current value of 100units and

20

child 230b has an associated current value of 200units. The current value associated
with parent 220b is 900units. As described above, child 230b (which is shared with
parent 220a) has an associated current value of 200units. Child 230c has an
associated current value of 300units and a child 230d has an associated current value

5   of 400units. The values associated with children 230 may be expressed in a vector as
follows:

$$\bar{x} = \begin{bmatrix} 100 \\ 200 \\ 300 \\ 400 \end{bmatrix}$$

10   In this example, the sum of the values of children 230 equals the value of their
respective parents 220. Therefore, the parent-child relationship matrix may be
expressed as follows:

$$R = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

15

Furthermore, assume that the variance matrix for children 230 is as follows:

$$\Sigma = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 20 & 0 & 0 \\ 0 & 0 & 30 & 0 \\ 0 & 0 & 0 & 40 \end{bmatrix}$$

20   Assuming that the forecasted value associated with parent 220a at time t is 400units
and the forecasted value associated with parent 220b at time t is 1000units, the values
allocated to children 230 at time t using Equation (3) above may be determined as
follows:

25

$$\bar{x}' = \bar{x} + \Sigma R^T (R \Sigma R^T)^{-1} (\bar{y} - R\bar{x})$$

$$\bar{x}' = \begin{bmatrix} 100 \\ 200 \\ 300 \\ 400 \end{bmatrix} + \begin{bmatrix} 10 & 0 \\ 20 & 20 \\ 0 & 30 \\ 0 & 40 \end{bmatrix} \begin{bmatrix} 0.039 & -0.008 \\ -0.009 & 0.013 \end{bmatrix} \left(\begin{bmatrix} 400 \\ 1000 \end{bmatrix} - \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 100 \\ 200 \\ 300 \\ 400 \end{bmatrix} \right)$$

$$\bar{x}' = \begin{bmatrix} 130.43 \\ 269.57 \\ 313.04 \\ 417.39 \end{bmatrix}$$

It should be noted that the sum of the allocated values for children 230 equal the forecasted values of their respective parents 220. Furthermore, these values may be compared to the values obtained using a proportional allocation technique. Using such a technique, the values of children 230a and 230b of parent 220a would be 133.33units and 266.67units, respectively. The values of the children 230b, 230c, and 230d of parent 220b would be 222.22units, 333.33units, and 444.44units, respectively. However, these values do not account for the variations in the values associated with each child 230 and thus likely to be less accurate than the values that are obtained above using the example method. Furthermore, the proportional allocation technique produces two different values for child 230b since the proportional allocation method is performed separately for each parent 220 and the allocation from each parent 220 produces a different result for the value to be associated with child 230b. These different results could then have to be reconciled. Therefore, system 10 is also advantageous in that it can simultaneously allocate values from multiple parents 220 to their children 230 (some of which may be common to two or more parents).

As can be seen from the above examples, system 10 provides for the allocation of data from parents to children in a hierarchical organization of data having one or more dimensions in a manner likely to be more accurate than with previous techniques. System 10 provides a number of advantages. For example, the representation of parent child relations using the parent-child relationship matrix **R** is

a flexible and general mathematical representation allowing greater flexibility and rigor in allocation from parents to children. Furthermore, when the quantities involved in an allocation are appropriate statistical quantities, the result from the allocation is statistically optimal.

5   Although particular examples are described above, the described techniques may be used for many other applications. For example, one advantage is the generality offered by the parent-child relationship matrix in handling and representing parent-child relations. The parent-child relationship matrix can have as many rows as there are parents whose values need to be allocated to their children. However, if

10   multiple parents do not have common children, computation may be simplified by separating such rows into different parent-child relationship matrices.

The most elementary case is a parent-child relationship matrix having a single row. The columns of the matrix represent the total number of children involved in a parent child relationship with a parent represented by the row of the matrix. Each

15   child typically has only one column regardless of the number of parents the child has. The value of an element in a row will typically be zero if the corresponding column is not a child of the parent represented by the row. A nonzero value indicates that the corresponding column is a child of the parent represented by the row. The nonzero value itself could be any number, so that any linear relationship could exist between a

20   set of children and their parent.

One example of a type of parent-child relationship is when the parent is equal to the sum of its children (an aggregation relation), as described above. In this case, each element of a row in the parent-child relationship matrix is zero if a child is not involved in the aggregation relation and is a one when the child is involved in the

25   aggregation. Another type of parent-child relationship is when the parent is the average of its children. In this case, each element of a row of the parent-child relationship matrix is zero when the corresponding child is not involved in averaging and is the fraction $1/n$ (where n is equal to the number of children of the parent) when the corresponding child is involved in the averaging. Yet another example of a

30   parent-child relationship is when the parent is a dollar value of a group of items and the children are quantities of each item contributing to the dollar value of the parent. In this case, each element of a row of the parent-child relationship matrix is zero when

the corresponding child is not involved in contributing to the dollar value represented by the parent. The value of the element is equal to the price of the corresponding child when the child is involved in contributing to the dollar value represented by the parent.

5   It should be noted that a row of the parent-child relationship matrix may represent a direct parent of the children involved in an allocation or an indirect parent. For example, the parent represented by a row may be a parent of another member that is a parent of the children involved in the allocation. Furthermore, although the values in the parent-child relationship matrices described above are numerical, the

10   values may also be semantic or have particular business meanings.

The flexibility offered by the parent-child relationship matrix is not restricted to cases where some or all of the values to be allocated are statistical quantities. For example, $\bar{x}$ may be zero or may be the result of other types of computations or user inputs. Similarly, Σ also may include either statistical quantities, more simplified

15   user-inputs, results of other non-statistical computations, or any other appropriate values. For certain cases, Σ may include the same values as $\bar{x}$, or be directly related to those values, along its diagonal and have off-diagonal elements equal to zero.

For example, $\bar{x}$ may be regarded as zero and Σ may be populated with values along its diagonal which are not all zero. In this case, system 10 could be used to

20   perform a proportional allocation of the desired values of the parents to the children, taking advantage of the ability of the method to handle multiple parents with shared children. A similar example is when it is desirable to allocate the difference between the current value of a parent and the desired value, according to a particular proportion among its children. In this case, not all elements of $\bar{x}$ may be regarded as

25   zero. The current values of $\bar{x}$ (or their functions, such as their square root or square) may be used as weights for an allocation and appear as the diagonal elements of Σ. Again, the advantage is the ability to allocate multiple parents to their children in a consistent fashion.

The above examples show that the parent-child relationship matrix and a

30   variety of choices for $\bar{x}$ and Σ allow for a flexible and generalized allocation scheme with respect to parent-child relations. Additional flexibility and rigor is obtained in

the allocation by using a variety of different types of values as the contents of Σ. As a example only, one can design a Σ matrix with the variances or a measure of the relative variation of the children along its diagonal. Further, unlike in the previous examples, the off-diagonal values of Σ can be non-zero and made equal to measures

5 of covariances or relative co-variation of each pair of children. This structure of Σ, when used in an allocation scheme, can account for relations between the children themselves. For example, when the value of one child i is higher, another child j may tend to be higher or lower to a degree specified by the quantity in the ith row and jth column of Σ.

10 Furthermore, the final allocated quantity may not be one that is explicitly produced by using Equation (3) presented above. For example, determining the final allocated quantity may involve selecting between and/or combining outputs obtained from different allocations using Equation (3). One reason for such selection and/or combination is that there may be uncertainties about the accuracy of the various

15 quantities involved in allocation. In such cases, it may be preferable to use alternative quantities in the allocation method and combine the results of the allocations in an appropriate manner (for example, by averaging the results) such that the final quantities after allocation might not be the result of applying the method to any one choice of input quantities. Similarly the output from one or more of the allocations

20 may be selected based on appropriate criteria.

Another problem that may arise during an allocation are inconsistencies or conflicts in relationships between parents and children or between parents. A conflict may arise when the underlying "degrees of freedom" for allocating a value are less than the number of rows in R. As an example, assume that there is a parent whose

25 value is to be allocated between two children, C1 and C2. The user may specify that C1 must have a particular fixed value; however, a particular fixed value can not then be specified for C2 since the parent value could not be allocated (there would not be enough degrees of freedom to allocate the parent value since both child values are fixed). When there are multiple parents being allocated to common children (possibly

30 in multiple dimensions) the relationships are much more complicated and the conflicts are harder to identify. However, the method described below can be employed to

determine the dependencies in an allocation set. Furthermore, the quantities computed are intermediate to the allocation method itself and thus there is no additional overhead for detecting the conflicts. Moreover, the method can be used to automatically resolve these conflicts by elimination without any user intervention,

5   when desirable. Alternatively, it is also possible to accept incremental modifications to the current allocation set when a conflict is detected (with no significant overhead), so that the user can specify what gets eliminated.

As described above, conflicts can occur when specifying various points (such as a parent or any other position where a user desires to specify a predetermined

10   value) in the hierarchies where certain predetermined values need to be fixed for allocation to their children. These conflicts result in conflicting requirements for the values at the target positions (for example, children), and cannot be resolved unless one or more of the conflicting requirements are eliminated. The cause of such conflicts is in the selection of the set of source positions (for example, parents) whose

15   effects have to be allocated to the target positions. Consider an N dimensional space spanned by the children and p dimensional space spanned by the parent-child relationship matrix R (where the p dimensional space is a subspace of the N dimensional space). Therefore, p can not be greater than N without introducing conflicts. When a source position for allocation is conflicting with a set of other

20   positions for allocation, the space spanned by the matrix R is degenerate in the sense that at least one of the rows of R is determined by a linear combination of the remaining rows (for example, if the row rank of R is less than the number of rows in R). Therefore, a conflict in allocation may be defined as a condition in which some source positions in the allocation set completely determine some other source

25   position(s) in the allocation set. Since the latter are completely dependent on the former, they are not available for change. Any change in their value(s) are the result of changes in the values of the other source positions which completely determine them.

This creates a linear dependency problem which may be resolved by removing

30   at least one of the rows of R (which may be each associated with a parent) involved in the dependency, in order for the rest of the requirements to be satisfied exactly (another approach to resolving conflicts might be to satisfy the requirements

approximately without excluding any requirement completely). Regardless of the allocation method used, analyzing the vectors of the parent-child relationship matrix R or, alternatively, the parent-parent relationship matrix $R\Sigma R^{T}$ (as described below) will enable the detection and resolution of conflicts in an allocation.

5   Particular embodiments of the present invention provide techniques for detection and resolution of conflicts in an allocation based on the properties of the matrix $R\Sigma R^{T}$ (or RR^{T}), as discussed above. According to these techniques, a conflict-free allocation set is a set having the matrix $R\Sigma R^{T}$ (or RR^{T}) with a non-zero determinant. The determinant is zero if a conflict is present. The basic approach

10   of the techniques is to compute the inverse of the relevant matrix using the pivoting method described below and, in the process, detecting the rows (or columns) of the matrix which are involved in a dependency with previously considered rows (or columns). When a dependency is detected, the position involved in the dependency is removed from the allocation scheme and the corresponding elements are removed

15   from the matrix, resulting in resolution of the conflict that was detected. The techniques described below may be implemented in any appropriate combination of software and/or hardware operating in association with one or more computers 20 of system 10.

Two operators referred to as *SWEEP* and *INVSWEEP* may be used in

20   association with determining the inverse. The result of *SWEEP* on all diagonal elements of a positive definite matrix is the inverse of the matrix. The effect of a single *SWEEP* is the removal of the influence of one row (or column) vector from the remaining vectors. One of the useful properties of the *SWEEP* operator is its ability to compute the determinant. The determinant may not be computed explicitly, with only

25   the rows (or columns) causing the determinant to be zero being detected and eliminated, thus producing a generalized inverse of the original matrix when its determinant is zero. The *INVSWEEP* operator is the inverse of the *SWEEP* operator so that *INVSWEEP* can be used to reintroduce the effect of a vector which has previously been swept.

30   There are several variants of the operator *SWEEP*. In the following methods, a variant is used that results in the negative inverse of the original matrix. One

27

advantage of this particular variant of *SWEEP* is that it is easy to undo the effect of the operator by using the given *INVSWEEP* operator; however, any other appropriate variant may be used. The particular variant of *SWEEP* that is used is as follows:

5   Let A be a symmetric matrix. Sweeping on the kth diagonal entry, $a_{k,k} \neq 0$, results in a new symmetric matrix $\hat{A} = (\hat{a}_{i,j})$ where,

$$\hat{a}_{k,k} = -\frac{1}{a_{k,k}}$$

$$\hat{a}_{i,k} = \frac{a_{i,k}}{a_{k,k}}$$

$$\hat{a}_{k,j} = \frac{a_{k,j}}{a_{k,k}}$$

$$\hat{a}_{i,j} = a_{i,j} - \frac{a_{i,k} a_{k,j}}{a_{k,k}}$$

for $i, j \neq k$

10

Inverse sweeping on the kth diagonal entry, $a_{k,k} \neq 0$, results in a new matrix $\hat{A} = (\hat{a}_{i,j})$ where

$$\hat{a}_{k,k} = -\frac{1}{a_{k,k}}$$

$$\hat{a}_{i,k} = -\frac{a_{i,k}}{a_{k,k}}$$

$$\hat{a}_{k,j} = -\frac{a_{k,j}}{a_{k,k}}$$

$$\hat{a}_{i,j} = a_{i,j} - \frac{a_{i,k} a_{k,j}}{a_{k,k}}$$

15   for $i, j \neq k$.

Three example techniques are presented below for inverting $R \Sigma R^T$ using the above operators and determining a set P of positions that are free from conflicts

(RR^T may be similarly inverted, if appropriate). The first technique computes the inverse when the matrix is full rank. When the matrix is rank-deficient, the technique eliminates one parent involved in each dependency detected and produces a generalized inverse of the matrix that can be used for allocation. However, in eliminating the dependencies of the parents, this technique does not expect user inputs and runs automatically. The second technique works like the first technique, except that when a dependency is detected, the technique allows for user input to eliminate a parent involved in the dependency and proceeds to produce a generalized inverse. These two techniques demonstrate the basic functioning of the method of determining a conflict-free set P of positions for allocation from a given set. Using aspects from first and second techniques, several levels of user interaction and automation may be implemented. The third technique presents an example of one such approach where the user may control the elimination of a subset of positions (as in the second technique) while the remaining positions are free for automatic elimination (as in the first technique). The outputs of all of these techniques can then be used for computing the allocation, as described above.

The first technique (automatic resolution of conflicts) may be implemented as follows:

Let $A = R\Sigma R^T$ (a $p \times p$ matrix), let δ be a very small number (such that δ is not zero, but as close to zero as is appropriate for computational purposes), and let P be a set of conflict-free positions. The following process may then be performed:

Initialize P as an empty set
For each i such that $1 \le i \le p$
{
   If $a_{i,i} > \delta$
   then
   {
     SWEEP on $a_{i,i}$
     Add position i to P
   }
   else
     Set all values in ith row and ith column to zero.
}

After the process is completed the set P will contain a set of positions that are consistent and free of conflicts. Furthermore, the matrix resulting from both the techniques will be a negative inverse of A which can be used in the allocation

5   computation as the quantity $\left(R\varSigma R^T\right)^{-1}$.

Similarly, the second technique (a technique for interactive resolution of conflicts) may be performed as follows:

10
```
Initialize P as an empty set
For each i such that 1 ≤ i ≤ p
{
        If a_{i,i} > δ
        then
        {
                SWEEP on a_{i,i}.
                Add position i to P
        }
        else
        {
                dependency ← TRUE
                repeat
                {
                Show P and i to user for elimination and get input k
                        such that a_{i,k} ≠ 0 or k=i
                Set all values in kth row and kth column to zero.
                If (k=i)
                        dependency ← FALSE
                else
                {
                        INVSWEEP on a_{k,k}
                        Remove position k from P
                        If a_{i,i} > δ
                        then
                        {
                                SWEEP on a_{i,i}.
                                Add position i to P
                                dependency ← FALSE
                        }
                } until dependency = FALSE
```

                    }
                }

As with the first technique, after the process is completed the set P will contain the set

of allocations that are consistent and free of conflicts. Furthermore, the matrix

resulting from both the techniques will be a negative inverse of A which can be used

in the allocation computation as the quantity $\left(R\Sigma R^T \right)^{-1}$.

The third technique (a technique for selectively controlled resolution of

conflicts) is a generalization of the two techniques above and allows the user to

control conflict resolution on a certain subset of the positions while allowing the

technique to automatically resolve on all other positions. This technique may be

implemented as follows:

Let $A = R\Sigma R^T$ (a $p \times p$ matrix), let δ be a very small number (such that δ is not zero, but as close to zero as is appropriate for the calculations), let P be the set of conflict-free positions, and let C contain the row numbers corresponding to the allocation positions that the user wants to control. The following process may then be performed:

Initialize P as an empty set
For each i such that $1 \le i \le p$
{
    If $a_{i,i} > \delta$
    then
    {
        SWEEP on $a_{i,i}$.
        Add position i to P
    }
    else
    {
        If $i \notin C$
        then
            Set all values in ith row and ith column to zero.
        else
        {
            dependency $\leftarrow$ TRUE
            repeat
            {

Show P and i to user for elimination and get input k such that $a_{i,k} \neq 0$ or $k=i$

Set all values in kth row and kth column to zero.

If ($k=i$)

> *dependency* $\leftarrow$ FALSE

else

{

> *INVSWEEP* on $a_{k,k}$
>
> Remove position k from P
>
> **If** $a_{i,i} > \delta$
>
> **then**
>
> {
>
> > *SWEEP* on $a_{i,i}$.
> >
> > Add position i to P
> >
> > *dependency* $\leftarrow$ FALSE
>
> }

}

} **until** *dependency* = FALSE

}

}

}

After the process is completed the set P will contain the set of allocations that are consistent and free of conflicts. Furthermore, as with the previous techniques, the matrix resulting from both the techniques will be a negative inverse of A which can be used in the allocation computation as the quantity $\left(R \Sigma R^T \right)^{-1}$.

FIGURE 7 illustrates an example method of detecting and resolving conflicts before an allocation. The method starts at step 300 where a rule set is selected for identifying and resolving conflicts associated with an allocation. For example, one of the three techniques described above may be selected for use (or any other appropriate technique may be selected) and/or other parameters associated with conflict detection and resolution may be selected. At step 302, the set P of conflict-free positions is initialized as an empty set and index i is set as the first parent or other position to be analyzed. System 10 determines at step 304 whether the total "weight" of position i (the total influence of the position) is greater than zero (or a very small number may be used to effectively represent zero for computational purposes). The total weight of position i is the value $a_{i,i}$ described above (when $A = R\Sigma R^T$ or RR^T).

If the total weight of position i is greater than zero (or greater than the very small number that effectively represents zero), then the influence of position i is removed from the other parents at step 306. As described above, this step may be carried out by performing a *SWEEP* on $a_{i,i}$ or using any other appropriate technique.

At step 308, position i is added to set P and system 10 determines at step 310 whether position i is the last position. If so, the method ends. If not, index i is set to the next position at step 312 and the method returns to step 304.

If the total weight of position i is zero (or not greater than the very small number that effectively represents zero), then the weight that position i shares with other positions (the "shared weight") is examined at step 314. For example, the shared weight $a_{i,k}$ (described above) identifies the relationship, if any, between position i and a selected position k. At step 316, system 10 may determine the subset of positions in set P with which position i has a relationship. A position k may have a relationship with position i if the shared weight $a_{i,k}$ is not equal to zero (or is greater in absolute value than the very small number used to effectively represent zero). A position k, which could be either i or a member from set P, is selected for elimination at step 318. The manner in which a position k is selected may differ based on the conflict resolution technique that is used. For example, if the automatic resolution technique is used, then the position k that is selected may simply be the last position i that was evaluated. If the technique for interactive resolution of conflicts is used, then the choice of position k may be left to the user. If the technique for selectively controlled resolution of conflicts is used, then the user may select a position k if the position i being evaluated is included in the set C (as described above). Alternatively, the rule set defined for identifying positions for elimination might impose a priority ordering on the positions involved in the conflict for identifying the positions for elimination.

At step 320, system 10 determines whether the selected position k is the current position i. If it is, then the method proceeds to step 310. If it is not, then the position k is a member of the set P and the effect of position k is reintroduced on the other positions at step 322. This step may be carried out by performing an *INVSWEEP* on $a_{k,k}$. Position k is removed from set P at step 324 and then all values

in the kth row and kth column of A to zero. The method then proceeds to step 106 where the influence of position i is removed from all other positions, as described above (for example, using a *SWEEP*), and position i is added to set P at step 308. The method may then proceed to step 310 where system 10 determines whether position i is the last position to be evaluated. As described above, if it is not, then the method proceeds to step 312. If position i is the last position, then the method ends. Using this example method, conflicts in an allocation set may be detected and resolved before the allocation is performed to prevent problems during the allocation. These conflicts may be detected and resolved even though complex hierarchical dependencies (that may lead to the conflicts) may exist between positions. Furthermore, certain quantities calculated during the conflict resolution process may be used during the allocation process, thus leading to computational efficiencies.

Although the present invention has been described with several embodiments, numerous changes, substitutions, variations, alterations, and modifications may be suggested to one skilled in the art, and it is intended that the invention encompass all such changes, substitutions, variations, alterations, and modifications as fall within the spirit and scope of the appended claims.